

Complementos de Bases de Dados 2025/2026



Licenciatura em Engenharia Informática

Projeto Prático – Época de Avaliação Contínua, Época Normal e Recurso

O projeto prático da disciplina de Complementos de Bases de Dados visa a aplicação de tópicos avançados de administração de bases de dados. O trabalho incidirá ao nível da BD e do SGBD e sobre a lógica de utilização dos dados para os requisitos colocados; **não se pretende o desenvolvimento das camadas da lógica aplicacional e apresentação.**

1 Descrição Geral

O grupo *Adventure*, do qual faz parte a empresa de material de ciclismo, *AdventureWorks*, está a ser alvo de uma reestruturação.

Uma das medidas adotadas pela administração passa por desenvolver um novo ERP que permita gerir, de forma integrada, todo o processo de venda dos seus produtos. Até ao momento a empresa gere toda a informação num ERP desatualizado e com recurso a ficheiros de Excel.

Com a implementação de um novo ERP torna-se necessário modelar e integrar, numa nova Base de Dados, os fragmentos de informação que foram exportados do ERP e das aplicações existentes que suportavam até à data a *AdventureWorks*.

Constitui-se anexo a este enunciado um conjunto de ficheiros (*dataset*), que contém os dados extraídos dos vários sistemas. No entanto, estes apresentam-se fracamente relacionados, carecendo de uma otimização, segundo as boas práticas de modelação e regras da normalização, com vista a providenciar um suporte eficiente das operações e ainda otimizar as intervenções de administração.

NOTA: Cada grupo receberá um dataset e uma ficha de grupo específicos para o seu trabalho. Deverá articular com o docente do laboratório a obtenção dos respetivos materiais.

2 Fase 1 – Modelação e Carregamento

2.1 Importação dos ficheiros “*legacy*”

Crie a base de dados “*AdventureWorksLegacy*” para onde deve importar toda a informação do sistema antigo constituinte dos ficheiros fornecidos.

2.2 Requisitos e Modelo de Dados

2.2.1 Modelo Entidade Relacionamento (MER)

Com base na análise das tabelas da base de dados “*AdventureWorksLegacy*” resultantes da importação dos ficheiros, deverá ser proposto um novo modelo entidade relacionamento (MER) com o respetivo diagrama contemplando:

- Informação de negócio;
 - Entidades de negócio inferidas da análise do *dataset* fornecido;

- Atender aos requisitos específicos para o seu grupo de trabalho de acordo com a ficha de grupo que lhe foi atribuída e acompanha o *dataset*, relativamente a:
 - Estrutura
 - Regras de Negócio
 - Campos Sensíveis,
- Gestão de utilizadores (de utilizadores de acesso à aplicação).
 - A autenticação do acesso dos clientes perante a aplicação será feita com recurso à conta de *email* e *password*;
 - No processo de recuperação da *password* de um cliente já existente, o sistema deverá gerar, automaticamente, uma nova *password* e enviar um e-mail ao utilizador com essa informação. **Nota:** no âmbito do projeto, poderá ser efetuada uma simulação de escrita numa tabela “sentEmails” com campos: destinatário, mensagem, *timestamp*) em vez de configuração de servidor de e-mail.;
 - Para poder recuperar a *password*, cada utilizador deve ter definida uma questão à qual tem de responder corretamente para efetuar a recuperação.

O diagrama ERD deverá ser feito com recurso à ferramenta ERD Plus.

2.2.2 Modelo Relacional

Uma vez caracterizado o MER (identificadas entidades, atributos e relacionamentos com respetiva cardinalidade e participação) e proposto o respetivo diagrama DER; este deverá ser convertido no modelo relacional, observando todos os requisitos a considerar e considerando uma normalização correspondente pelo menos à 3ª forma normal.

Na concretização do desenvolvimento do modelo relacional deverá considerar as opções na definição pertinente dos tipos de dados, *default values*, *constraints* (*primary key*, *foreign key*, *unique* e *check*).

2.3 Layout da BD

A definição do *layout* da nova base de dados normalizada “AdventureWorks”, a ser justificado no relatório, deverá suportar-se num conjunto de dados, nomeadamente:

1. Espaço ocupado por registo de cada tabela;
2. Espaço ocupado por cada tabela, com o número atual de registos;
3. Tipo de acesso mais frequente (apenas de leitura ou leitura e escrita).

A definição lógica de uma configuração possível para os *Filegroups* com o respetivo tipo, que deverá ser claramente informada pelo levantamento supra enumerado.

Deverá criar explicitamente o código DDL para implementação do modelo em SQL Server.

2.4 Migração da informação

A informação constante da base de dados “AdventureWorksLegacy” deverá ser migrada para a nova base de dados “AdventureWorks”. A migração deverá ocorrer sem perda de informação.

Defina os scripts necessários para a migração dos dados antigos para a nova base de dados.

Verificação da qualidade dos dados

De modo a verificar a conformidade da migração efetuada, aplique às duas bases de dados as seguintes *queries*:

1. Nº de “*Products*”;
2. Nº de “*Sales*”;
3. Total de vendas por “*Customer*”;
4. Total monetário de vendas por ano;
5. Total monetário de vendas por ano e por “*Product*”.
6. Resposta à *query* específica do grupo fornecida na ficha de grupo

Notas:

- Deve ser contabilizada uma única venda (“*Sales*”), mesmo que esta contenha vários produtos;
- Valor monetário obtido por “*OrderQuantity*UnitPrice*”;
- O ano deve ser retirado da coluna “*OrderDate*”.

2.5 Programação

Deverá considerar o desenvolvimento de *triggers*, *stored procedures* e *functions* para:

- Apoio à migração dos dados
- *Stored procedures* e/ou *functions* utilitárias de apoio ao cálculo e/ou visualização da *queries* de verificação da qualidade dos dados
- Apoio à implementação dos requisitos complementares, nomeadamente:
 - Gestão de utilizadores:
 - Editar, Adicionar e Remover utilizador;
 - Recuperar Password (Questões e Resposta e verificação).
 - Visualização da informação de uma determinada compra de um cliente (parâmetros de entrada: data da venda e identificador de cliente).

Tratamento de Erros

Todo o código desenvolvido no projeto deverá fazer as validações necessárias e incluir tratamento de erros. O tratamento de erros deve ser gerido de forma centralizada. Na ocorrência de um erro deverá ser enviada uma mensagem amigável ao utilizador e criado um *log* do erro que identifique o erro gerado, o utilizador (MS SQL Server) e a informação temporal relativa à ocorrência do evento.

2.6 Catálogo/Metadados

Apoio à monitorização

Deverão ser criados, para efeitos de apoio à monitorização, os seguintes objetos na BD:

- *stored procedure* “*sp_dbstatistics*” que registe na tabela “*dbStatistics*”, para cada tabela da base de dados, o seu número de registos, o espaço ocupado e data da última atualização. Deverá manter histórico dos resultados das sucessivas execuções da *sp*.

3 Fase 2 – Otimização e Administração

3.1 Índices

Considerando que o departamento de marketing necessita de fazer algumas análises e relatórios sobre as vendas, implemente o seguinte conjunto de *queries*:

1. Pesquisa de vendas por cidade. Deve ser retornado o nome da cidade, o código do estado e o total de vendas;
2. Pesquisa de produtos associados a vendas com valor total monetário superior a 1000;
3. Número de produtos vendidos por categoria por ano.

Notas: Cidades com o mesmo nome, mas de diferentes estados, deverão ser consideradas distintas.

Com vista à otimização da execução das consultas propostas, defina, justificadamente, os índices pertinentes. Inclua no relatório a justificação apresentada - incluir o script SQL na lista de ficheiros a apresentar.

SQL Profiler e Tuning Advisor

Contraste os planos de execução das *queries* anteriores, sobre a base de dados, que otimizou e as *queries* equivalentes sobre a base de dados original (sem normalização).

No caso da nova base de dados, devem ser apresentados os planos de execução com e sem índices. Inclua no relatório o comentário ao que observa.

3.2 Backup e Recuperação

Defina e implemente uma estratégia de *backup* e recuperação para o Sistema de Informação, concretizando no relatório aspetos como:

1. Modelo de Recuperação;
2. Tipo de *backup* (integral, diferencial e *log*). assuma e apresente os pressupostos relativos à carga do sistema, de forma a distribuir justificadamente o tipo de backup e definir a rotação dos dispositivos de armazenamento.
3. Simule um cenário de ocorrência de “*crash*” da base de dados e apresente a sequência de recuperação.

3.3 Níveis de acesso à informação

Defina os objetos necessários e os privilégios de acesso aos objetos da base de dados, para os seguintes tipos de utilizadores:

1. Administrador: Tem acesso a toda a informação;
2. SalesPerson: Tem acesso total às tabelas de suporte às vendas e apenas acesso em modo de consulta às restantes tabelas;
3. SalesTerritory: Pode consultar a informação relativa ao seu território. Considere apenas o território “*Rocky Mountain*” (nota: podem ser criadas views auxiliares).

Considere a necessidade de implementar, na base de dados, as contas de acesso diferenciadas para cada um dos utilizadores anteriormente identificados.

3.4 Encriptação

Implemente o código necessário à encriptação, ou *hash* dos campos sensíveis que lhe foram comunicados na ficha de grupo. Considere ainda a ofuscação dos campos relativos às perguntas e respostas para a recuperação de *password*.

Inclua no relatório a justificação da escolha da metodologia escolhida para cada um dos campos.

3.5 Controlo de transações

Defina os níveis de isolamento para o cenário de potencial conflito que lhe é apresentado na sua ficha de grupo.

Descreva no relatório a solução implementada para cada um dos cenários identificados.

Preveja uma simulação/demonstração representativa.

3.6 NoSQL - MongoDB

Com o objetivo de permitir ao departamento de Vendas realizar relatórios de balanço sobre as compras efetuadas, o grupo *Adventure*, vai disponibilizar num *site* alguma informação. Contudo, de modo a não adicionar carga ao sistema atual, o *site* vai ser implementado num sistema autónomo, tendo por base o SGBD NoSQL, MongoDB.

A informação a contemplar na base de dados MongoDB será a solicitada na sua ficha de grupo.

Esta etapa inclui as seguintes tarefas:

- Criar a base de dados *AdventurWorksWeb*;
- Implemente as “*collections*” necessárias para suportar a informação disponibilizada no site;
- Defina em SQL Server as consultas necessárias para popular a base de dados *AdventurWorksWeb* ;
- Importe para MongoDB os resultados da alínea anterior;
- Defina 3 comandos/*queries* MongoDB que cumpram:
 1. Resultado excluindo explicitamente valores omissos e ordenado
 2. Envolve uma filtragem
 3. Envolve uma agregação

4 Regras de Execução

O não seguimento de qualquer das regras elencadas nas secções seguintes pode ser motivo de anulação do projeto.

4.1 Grupos de Trabalho e Acompanhamento

O projeto deverá ser desenvolvido por grupos de 2 alunos. Só em casos excecionais e devidamente justificados poderá ser executado individualmente, sob parecer positivo do responsável da UC (Prof. João Portelinha Santos). Os elementos do grupo de trabalho têm de pertencer à mesma aula de laboratório e serão acompanhados e avaliados pelo respetivo docente.

Para o acompanhamento estarão disponíveis as aulas de laboratório e os horários de dúvidas de acordo com o publicado no *moodle*. Poderão ainda ser utilizados fóruns de discussão para o efeito que estão disponíveis no *moodle*.

4.2 Calendário e Entregas

O projeto rege-se pelo seguinte calendário:

Data	Descrição
Até às 23h55 do dia 07/11/2025	Entrega da Fase 1
Até às 23h55 do dia 19/12/2025	Entrega da Fase 2
Data do exame da época normal	Entrega Projeto (Fase 1 + Fase 2)
Data do exame da época de recurso	Entrega Projeto (Fase 1 + Fase 2)

A entrega deve ser realizada através da plataforma *Moodle*, em *link* disponível para o efeito (discriminado por docente/turno de laboratório).

Deverá compor a entrega um ficheiro .ZIP com o seguinte conteúdo:

- Relatório
 - Documentando o projeto e justificações para as principais decisões tomadas em todas as secções de requisitos relativos a cada uma das fases de desenvolvimento.
 - Será disponibilizada no moodle a *template* para execução do relatório.
- Scripts desenvolvidos
 - O código implementado deverá ser devidamente comentado, **tendo no cabeçalho (interior) de cada ficheiro, a identificação dos elementos do grupo e descrição geral do conteúdo do ficheiro**

O nome do ficheiro ZIP segue a seguinte nomenclatura: NumeroAluno1_numeroAluno2.zip

Entregas fora de prazo serão alvo de uma penalização de 1 valor por hora nas 10 horas subsequentes à entrega, findas as quais o projeto será anulado.

Todos os materiais entregues (nomeadamente relatório e scripts) serão executados no software anti-plágio da instituição.

Incumprimentos detetados serão alvo de análise com possível instauração do processo correspondente.

4.3 Avaliação

Discussões marcadas segundo escala publicada pelo docente

Cada grupo deverá na sua hora ter tudo preparado para realizar a demonstração e responder às questões do docente

Durante a discussão as questões podem ser dirigidas a cada um dos elementos do grupo. Consequentemente podem resultar notas diferentes para os elementos do grupo de trabalho. Código obtido a partir de fontes externas, sem que as mesmas sejam devidamente referenciadas, leva à anulação do trabalho.

Grelha de Avaliação

Fase 1

Critério de passagem absoluto: min de 7 valores na componente Requisitos Específicos

Componente	Valor
Modelo	20%
Layout	5%
Migração de Dados e <i>queries</i> de validação	25%
Programação e Tratamento de Erros	15%
Metadados	10%
Requisitos Específicos	15%
Relatório e Apresentação	10%
Total	100%

Fase 2

Critério de passagem absoluto: min de 7 valores na componente Requisitos Específicos

Componente	Valor
Correções e otimização da fase 1	10%
Índices	15%
Backup e Recuperação	10%
Níveis de acesso	10%
Encriptação	5%
Controlo de transações	10%
MongoDB	15%
Requisitos Específicos	15%
Relatório e Apresentação	10%
Total	100%

Fim do enunciado